「1」にする。

このように、受信した圧縮コードに基づいて圧縮コードのまま長さ（ランレングス）をチエックして正しいラインか否かを判定し、正しいラインであれば受信した圧縮コードをメモリに蓄積することにより、圧縮コードを一旦伸長してランレングスをチエックして正しいラインか否かを判定する場合に比べて判定処理を高速で行なうことができるようになる。

このようにしてＦＡＸ画情報蓄積エリアに蓄積された後受信印刷指示が与えられることによって、蓄積されたＭＨコードをデコードしてプリンタ・ライン・ビット・プレーンＰＬＢＰ（これについては後述する）を作成し、このプリンタ・ライン・ビット・プレーンＰＬＢＰのデータをプリンタに出力する。

そこで、まずＦＡＸ画情報蓄積エリアに蓄積された圧縮コードをバイトあるいはワード単位で取出してデコードし、プリンタの１ライン分のイメージデータを作成する処理について説明する。

このイメージデータの作成処理では、圧縮コードをデコードして１行（１ライン）分のイメージデータを作成して、メモリ（このメモリを上述した「プリンタ・ライン・ビット・プレーンＰＬＢＰ」と称する）に格納する。

この場合、この実施例におけるプリンタは１行（１ライン）が２４×１４４０ドット構成であるのに対して、ファクシミリ装置においては１行の長さが１７２８ドットであるため、第１３図に示すようにプリンタ・ライン・ビット・プレーンＰＬＢＰよりも実際にデコード処理で得られるビット・プレーンの方が両端が各々１４４ドット分大きくなる。したがつて、この１４４ドット分はデコードしたデータをそのままプリンタ・ライン・ビット・プレーンＰＬＢＰに展開して印字したときには欠けることになるが、この点については具体的説明は省略するがビット変換処理を行なうことによつて大幅な欠落を回避できる。

そして、ここでは、プリンタ・ライン・ビット・プレーンＰＬＢＰのエリアとしてはこの実施例のプリンタ以外のプリンタを使用することも考慮して、ファクシミリ装置の幅に合わせて、１つのエリアが１７２８×２４（５１４８バイト×３）である（このエリアを「プレーン」と称する）３個のプレーンからなるプリンタ・ライン・ビット・プレーンＰＬＢＰを２個使用する。なお、プリンタ・ライン・ビット・プレーンＰＬＢＰを２個持つのはデコード処理とプリンタ出力処理とを並行動作できるようにするためである。

この場合、実際のプリンタ・ライン・ビット・プレーンＰＬＢＰのためのメモリ空間は例えば第１４図に示すようなメモリ空間としている。これは、送信モードにおけるビット・マップ展開で説明したように、プリンタへのイメージデータの転送フオーマットとファクシミリ装置との間の転送フオーマットが異なるので、送信時とは逆にファクシミリ装置から受信した１ドットラインのデータをプリンタに転送する際のフオーマット合わせて格納するためである。

そこで、デコード及びプリンタ・ライン・ビット・プレーンＰＬＢＰの作成処理について第１５図を参照して説明する。

この作成処理においては、受信終了処理テーブルから其情報を得た後、プリンタ・ライン・ビット・プレーンカウンタＰＬＮＣＮＴを「０」にリセットし、ドットライン数（０～２４－１）を示すためのライン・ホリゾンタル・ポインタＬＩＮＨＺＰを「０」にリセットする。

その後、更に１行（２４ドットライン）が全て白（０）であることを示す、つまり当該ラインには黒（１）が存在することを示すためのプリンタ・ライン・ビット・プレーン・ブランク・イグジストフラグＰＬＮＢＬＫを「０」（すべて白の状態）にセットし、プリンタ・ライン・ビット・プレーンＰＬＢＰをクリアする。

そして、１ドットラインのビット位置を示すためのライン・ビット・ポインタＬＩＮＢＴＰを「０」にリセットした後、白（ホワイト）圧縮コードをデコードするホワイトデコード処理に移行する。

このホワイトデコード処理でＥＯＬ及びエラーＥＯＬのいずれもを検出しなければ、ライン・ビット・ポインタＬＩＮＢＴＰに圧縮コードのランレングス（ＲＬ）分だけ加算し、ＥＯＬをカウントするＥＯＬカウンタを「０」にリセットした後、黒（ブラック）圧縮コードをデコードするブラックデコード処理に移行する。

このブラックデコード処理でＥＯＬ及びエラーＥＯＬのいずれも検出しなければ、当該ラインには黒があるのでプリンタ・ライン・ビット・プレーン・ブラック・イグジットフラグＰＬＮＢＬＫを「１」にセットする。

その後、ライン・ビット・ポインタＬＩＮＢＴＰが「１７２８」以上か否かを判別し、ライン・ビット・ポインタＬＩＮＢＴＰが「１７２８」未満であれば、所定の位置に「１」（黒）を配置するビットセット処理をした後、ライン・ビット・ポインタＬＩＮＢＴＰに圧縮コードのランレングス（ＲＬ）分だけ加算して、ランレングスから－１（ＲＬ－１）してその値が「０」か否かを判別し、（ＲＬ－１）＝０でなければ再度ライン・ビット・ポインタＬＩＮＢＴＰが「１７２８」以上か否かの判別処理に戻ってビットセットを繰返す。

そして、ライン・ビット・ポインタＬＩＮＢＴＰが「１７２８」以上になったとき又は（ＲＬ－１）＝０になったときには、１ドットラインの最終ビットに達しているのでそのままホワイトデコード処理に戻る。

このように、ここでは圧縮コードをデコードしてランレングスを求め、白ＭＨコードについては単にカウンタを進め、黒ＭＨコードについてのみそのランレングスで示されるビット分だけプリンタ・ライン・ビット・プレーンＰＬＢＰの所定のビットに「１」（黒）を配置する処理を行なう。それによって、白についてはビット配置を行なわなくてすみ、前述した送信処理の場合と同様に処理速度の高速化を図ることができる。

そして、ホワイトデコード処理又はブラックデコード処理においてＥＯＬを検出したときには、ＥＯＬカウンタをインクリメント（＋１）した後、またホワイトデコード処理又はブラックデコード処理においてエラーＥＯＬを検出したときにはそのまま、ＥＯＬカウンタのカウント値が「２」すなわちＲＴＣ（頁の終り）か否かを判別する。

このとき、ＥＯＬカウンタのカウント値が「２」でなければ、つまり頁の終りでなければ、印字ピッチが3.85ライン／mmか7.7ライン/mmか（3.85／7.7）かを判別して、印字ピッチが3.85ライン／mmのときにはライン・ホリゾンタル・ポインタＬＩＮＨＺＰを「＋２」し、印字ピッチが7.7ライン／mmのときにはライン・ホリゾンタル・ポインタＬＩＮＨＺＰを「＋１」する。なお、3.85ライン/mmのときにライン・ホリゾンタル・ポインタＬＩＮＨＺＰを「＋２」するのは後述するようにビットセット処理で7.7ライン/mmに合わせるために２ドットライン分同じビットセットを行なうためである。

そして、ライン・ホリゾンタル・ポインタＬＩＮＨＺＰが「２４」か否かつまり２４ドットライン（１行）分のデコードが終了したか否かを判別して、２４ドットライン分のデコードが終了していなければ再度ライン・ビット・ポインタＬＩＮＢＴＰを「０」にリセットしてデコードを繰返し、２４ドットライン分のデコードが終了していれば、プリンタ・ライン・ビット・プレーン・ブラック・イグジット・フラグＰＬＮＢＬＫが「１」か否かを判別する。

このとき、プリンタ・ライン・ビット・プレーン・ブラック・イグジット・フラグＰＬＮＢＬＫが「１」であれば、プリンタ・ライン・ビット・プレーンＰＬＢＰに展開したイメージデータをプリンタに出力し、「１」なければつまり「０」であれば、その行はすべて白であって単にラインフィード（改行）を行なえば足りるので、プリンタヘッドコントロールコマンド「ＬＦ」をプリンタに出力する。

その後、プリンタ・ライン・ビット・プレーン・カウンタＰＬＮＣＮＴをインクリメント（＋１）して、プリンタ・ライン・ビット・プレーン・カウンタＰＬＮＣＮＴのカウント値が最大値（ＭＡ

Xに等しいか否か、すなわち1頁の最大印字行数になつたか否かを判別する。

このとき、プリンタ・ライン・ビット・プレーン・カウンタPLNCNTのカウント値が最大値でなく1頁の最大行数になつていなければ、当該用紙に未だ印字できるので、プリンタ・ライン・ビット・プレーン・ブラック・イグジット・フラグPLNBLKが「1」か否かを判別する。

そして、プリンタ・ライン・ビット・プレーン・ブラック・イグジット・フラグPLNBLKが「1」のときにはプリンタ・ライン・ビット・プレーンPLBPに「1」が配置されているので、ライン・ホリゾンタル・ポインタLINHZPを「0」にリセットし、更にプリンタ・ライン・ビット・プレーン・ブラック・イグジット・フラグPLNBLKを「0」にリセットしてプリンタ・ライン・ビット・プレーンPLBPをクリアする処理に戻る。

また、プリンタ・ライン・ビット・プレーン・ブラック・イグジット・フラグPLNBLKが「1」でなければ、プリンタ・ライン・ビット・プレーンPLBPに「1」が配置されていないので、そのままライン・ビット・ポインタLINBTPを「0」にリセットする処理に戻る。

このようにデコードした行についてプリンタ・ライン・ビット・プレーンPLBPに「1」を配置したか否かを保持しておくことによつて、印字動作の高速化を図れると共に、ビット展開の処理の高速化を図ることができる。

なお、EOLカウンタEOLCNTのカウント値が「2」になつたとき、すなわちRTCを検出したときにはプリンタ・ライン・ビット・プレーン・ブラック・イグジット・フラグPLNBLKが「1」か否かを判別して、「1」であれば上述したように黒（1）があるのでイメージデータ（最終のイメージデータである）をプリンタに出力した後、また「1」でなければ全て白であるのでそのままヘッドコントロールコマンド「FF」（改頁）をプリンタに出力する。また、プリンタ・ライン・ビット・プレーンカウンタPLNCNTのカウント値が最大値「MAX」になつたときにも当該用紙には最早印字できないので、ヘッドコントロールコマンド「FF」（改頁）をプリンタに出力する。

次に、この作成処理におけるビットセット処理について第16図を参照して説明する。

このビットセット処理においては、まずライン・ビット・ポインタLINBTP及びライン・ホリゾンタル・ポインタLINHZPで示すビットのセット指示を受ける。

そして、ライン・ホリゾンタル・ポインタLINHZPのビットD_2，D_1（b_2，b_1）で示されるプリンタ・ライン・ビット・プレーンPLBPを構成する3個のバッファの内のいずれかのバッファ（第14図参照）を選択し、ライン・ビット・ポインタLINBTPと選択したバッファから実アドレスを得る。

その後、ライン・ホリゾンタル・ポインタLINHZPのビットD_2，D_1，D_0から6ビット位置を得て、このビット位置に黒（1）をセットする。なお、このとき、印字ピッチが3.85ライン／mmのプリンタでは、ビットD_0，D_1，D_2+1のビットについても黒を配置して同じドットラインのデータを2ドットライン分生成する。

次に、プリンタへの出力処理について第17図を参照して説明する。このプリンタ出力処理タスクはデコード処理タスクからのメッセージ（例えば「メッセージプリンタコマンドMSGPRC」とする）を受けて起動され、処理の終了時にデコード処理タスクに対してメッセージ（例えば「メッセージプリンタレスポンスMSGPRR」とする）を返送するものとする。

このプリンタ出力処理では、まずプリンタがノットビジイ状態（レディ状態）か否かを判別して、プリンタがノットビジイ状態でなく使用中であればブザーを鳴らしてメッセージプリンタレスポンスMSGPRRとして「エラー」を返送する。

これに対して、プリンタがノットビジイ状態であれば、プリンタに対して初期化コマンド「ESC，R」を送出して、プリンタの初期設定（6L

ＰＩ，ホームポジション，５０ＣＰＳ，黒とする）をさせる，

そして、プリンタに出力するデータがイメージデータか否かを判別し、イメージデータであれば、イメージデータコマンド「ＥＳＣ，％，１，ｎ１，ｎ２」（ｎ１，ｎ２でイメージデータの長さを示す）を出力して、イメージデータを出力してイメージデータを印字させ、キャリッジリターンコマンド「ＣＲ」及びラインフィードコマンド「ＬＦ」を出力して印字ヘッドを次行先頭位置に位置させた後、メッセージプリンタレスポンスＭＳＧＰＲＲとして「レディ」を返送し、メッセージプリンタコマンドＭＳＧＰＲＣを受けたときに再度イメージデータか否かの判別処理に戻る，

また、プリンタに出力するデータがイメージデータでなければ、改行コマンド「ＬＦ」か否かを判別し、改行コマンド「ＬＦ」であれば、改行コマンド「ＬＦ」を出力した後、メッセージプリンタレスポンスＭＳＧＰＲＲとして「レディ」を返送し、メッセージプリンタコマンドＭＳＧＰＲＣを受けたときに再度イメージデータか否かの判別処理に戻る，

さらに、プリンタに出力するデータが改行コマンド「ＬＦ」でなければ、改頁コマンド「ＦＦ」か否かを判別し、改頁コマンド「ＦＦ」であればキャンセルコマンド「ＣＡＮ」及び改頁コマンド「ＦＦ」を出力した後、メッセージプリンタレスポンスＭＳＧＰＲＲとして「レディ」を返送してこの処理を終了し、また改頁コマンド「ＦＦ」でなければメッセージプリンタレスポンスＭＳＧＰＲＲとして「エラー」を返送してこの処理を終了する，

次に、受信データ（圧縮コード）に基づいて用紙サイズを判定する用紙サイズ判定処理について第１８図乃至第２０図を参照して説明する，

まず、この実施例で使用している１０″プリンタでは、１ラインのドット数が前述したように、１４４０程度であり、実際には

Ａ４サイズ…１８６２ドット

Ｂ４サイズ…１４４０ドット

を採用している，つまり、第１８図に示すように、Ｂ４サイズでは１４４０ドット全部を使用するが、Ａ４サイズでは両端の各々３９ドット（合計７８ドット）は使用しない，

また、前述したようにファクシミリ装置では１ラインが１７２８ドットであるのに対してプリンタは１ラインが１４４０ドットであり、第１９図に示すように送信時に両端に各々１４４ドットの白ランを付加するので、受信ＦＡＸ画情報の内の両端の１４４ドットはプリンタの１ラインをはみでることになる，

そこで、ＦＡＸ画情報（圧縮コード）を受信してデコードするときに、各ラインの最初の白ランの長さの内での最小値と、各ラインの最後の黒ランの位置の内での最大値とを検出し、この検出した最小値が「１４４＋３９＝１８３」ドット位置より小さいとき、又は最大値が「１４４＋３９＋１３６２＝１５４５」ドット位置より大きいときにはＢ４サイズと判定し、そうでなければＡ４サイズと判定してＦＡＸ画情報と共に蓄積して、印刷時にその判定したサイズを表示するようにしている，

この処理について第２０図を参照して説明すると、まず左側の最小白ランポジションを格納するレジスタＬＰＯＳ及び右側の最大黒ランポジションを格納するレジスタＲＰＯＳを「０」にセット（クリア）した後、受信データ（圧縮コード）からＥＯＬをサーチする，

その後、１ライン中のデコード結果のランポジションをカウントするためのカウンタＲＵＮＣＮＴ及び１ライン中のデコード結果の黒の右側の最大ポジションをカウントするためのカウンタＬＲＵＮＣＮＴを「０」にセット（クリア）した後、受信データをデコードしてランレングスのビット数を求める，なお、これ等のカウンタＲＵＮＣＮＴ及びカウンタＬＲＵＮＣＮＴのカウント値は１ライン中でのテンポラリイ値である，

そして、ランレングスのビット数（以下「ＲＵＮ」と称する）がレジスタＬＰＯＳの値より小さい（ＲＵＮ＜ＬＰＯＳ）か否かを判別して、ＲＵ

ＮくＬＰＯＳであればＲＵＮをレジスタＬＰＯＳにセットする。その後、ＲＵＮにカウンタＲＵＮＣＮＴの値を加算した値を再度カウンタＲＵＮＣＮＴにセットする。

その後、次の受信データをデコードし、ＥＯＬか否かを判別する。このとき、ＥＯＬでなければ、ＲＵＮにカウンタＲＵＮＣＮＴの値を加算した値を再度カウンタＲＵＮＣＮＴにセットした後、黒ランか否かを判別し、黒ランであればカウンタＲＵＮＣＮＴの値をカウンタＬＲＵＮＣＮＴにセットした後、また黒ランでなければそのまま、次の受信データのデコード処理に戻る。

そして、受信データがＥＯＬになったときには、カウンタＲＵＮＣＮＴの値が「１７２８」か否かを判別し、カウンタＲＵＮＣＮＴ＝１７２８であれば、カウンタＬＲＵＮＣＮＴの値がレジスタＲＰＯＳの値より大きい（ＬＲＵＮＣＮＴ＞ＲＰＯＳ）か否かを判別し、ＬＲＵＮＣＮＴ＞ＲＰＯＳのときにはカウンタＬＲＵＮＣＮＴの値をレジスタＲＰＯＳにセットする。これに対して、カウンタＲＵＮＣＮＴ＝１７２８でなければエラー処理をする。

その後、１頁が終了したか否かを判別して、１頁が終了していなければカウンタＲＵＮＣＮＴを「０」にクリアする処理から再度上述した処理を行なう。このようにして、レジスタＬＰＯＳには１頁の各ラインの内で白ランポジションの最小値を格納し、レジスタＲＰＯＳには１頁の各ラインの内で黒ランポジションの最大値を格納する。

そして、１頁が終了したときには、レジスタＬＰＯＳの値が「１８３」（１４４＋３９）より小さい（ＬＰＯＳ＜１８３）、つまり左側の最小白ランポジションが「１８３」より小さいか否かを判別し、レジスタＬＰＯＳの値が「１８３」より小さいときにはＢ４サイズと判定し、またレジスタＬＰＯＳの値が「１８３」より小さくなければレジスタＲＰＯＳの値が「１５４５」（１４４＋３９＋１３６２）より大きい（ＲＰＯＳ＞１５４５）、つまり右側の最大黒ランポジションが「１５４５」より大きいか否かを判別し、レジスタＲＰＯＳの値が「１５４５」より大きいときにはＢ４サイズと判定し、更にレジスタＲＰＯＳの値が「１５４５」より大きくなければＡ４サイズと判定する。

なお、この処理は相手先が自己と同等の通信制御装置であって１８０ＤＰＩで印字データを送ってくる場合の例である。

そして、このようにして得られた用紙サイズを印刷時に表示することによってオペレータに適切な用紙のセットを促し、あるいは複数サイズの用紙をセット可能な例えばオートシートフィーダを搭載したプリンタであれば自動的に適切な用紙を選択することによって、送信側に応じた適切な用紙を使用してプリンタで印刷することができ、データの欠落等を生じない。

なお、上記実施例においては、ホスト側からプリンタに対する印字データがイメージデータで転送される例について述べたので、通信制御装置自体には文字コードを文字パターン（イメージデータ）に変換するキャラクタジェネレータを備えていないが、ホスト側からの印字データが文字コードで送られてくるときには通信制御装置内にキャラクタジェネレータを備えればよく、またこの内部キャラクタジェネレータの使用を選択する選択スイッチを設ければホスト側がイメージデータ及び文字コードのいずれを使用するものであっても接続できる。

また、上記実施例においては、この発明をプリンタに接続してファクシミリ装置から受信する通信制御装置について述べたが、これに限らず圧縮コードを受信するすべての通信端末装置に実施することができる。

さらに、プリンタの機能も上記実施例で述べたものに限られないことも勿論である。

効　果

以上説明したように、この発明によれば、適切な用紙を使用することができる。

4.図面の簡単な説明

第１図はこの発明を実施した通信制御装置の要部を機能的に示すブロック図。

第２図は同じくその通信制御装置を備えた情報処理システムの一例を示す構成図、

第３図は同じくその通信制御装置の具体的構成を示すブロック図、

第４図は同じくホスト側から入力されるコマンドを解析するコマンド解析処理を示すフロー図、

第５図及び第６図は同じくホスト側からのデータ転送の説明に供する説明図、

第７図及び第８図は同じくワークメモリへのビット・マップ展開の説明に供する説明図、

第９図は同じくそのワークメモリの説明に供する説明図、

第１０図は同じくビット・マップ展開処理の一例を示すフロー図、

第１１図及び第１２図は同じくラインチェック処理の一例を示すフロー図及びその具体的説明に供する説明図、

第１３図及び第１４図は同じくデコード処理の説明に供する説明図、

第１５図乃至第１７図は同じくデコード処理，ビットセット処理及びプリンタ出力処理を示すフロー図、

第１８図及び第１９図は同じく用紙サイズ判定処理の説明に供する説明図、

第２０図は同じく用紙サイズ判定処理の一例を示すフロー図である。

３…通信制御装置　　１３…通信手順制御部

１４…デコードラインチェック部

１５…ＰＡＸ画情報蓄積部

１６…印刷ドット領域検出部

１７…受信印刷指示部　１８…紙サイズ表示部

１９…デコード処理部

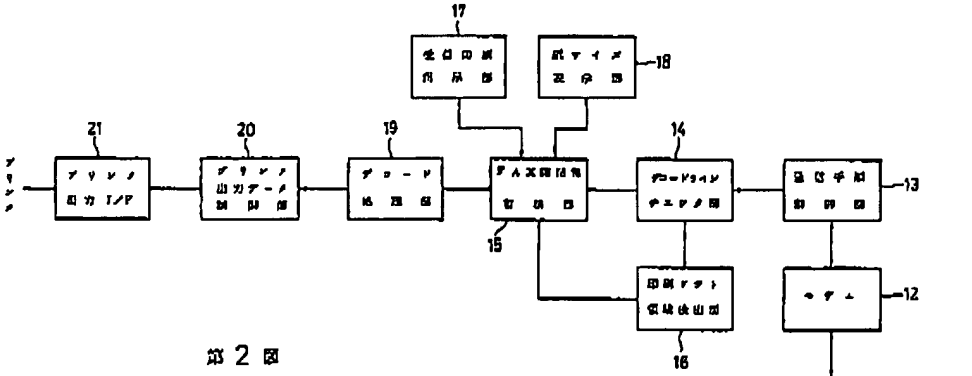
２０…プリンタ出力データ制御部

２１…プリンタ出力インタフェース

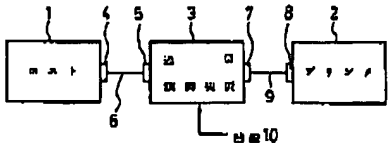
出願人　株式会社　リ　コ　ー

代理人　弁理士　大　澤　　敬

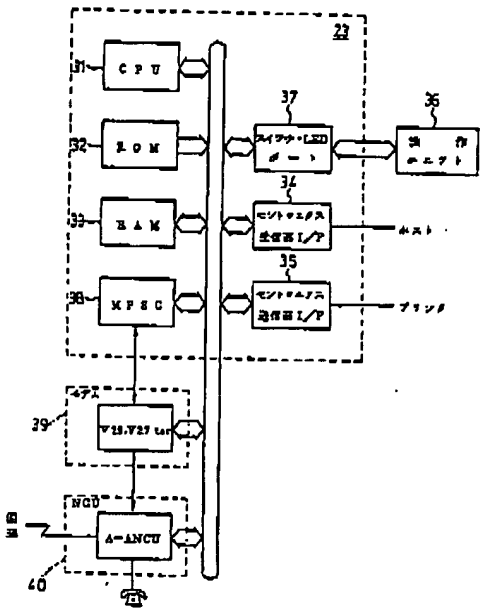
第１図



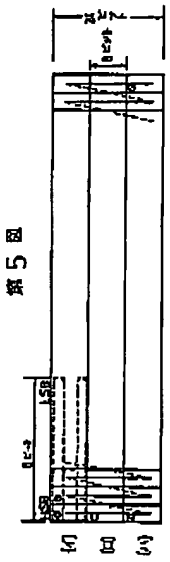
第２図



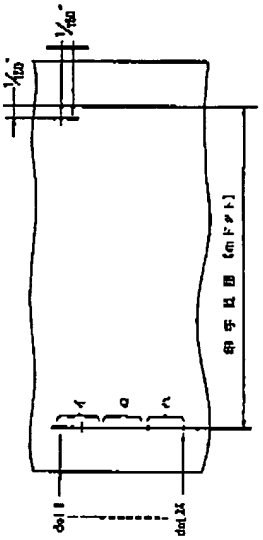
第3図



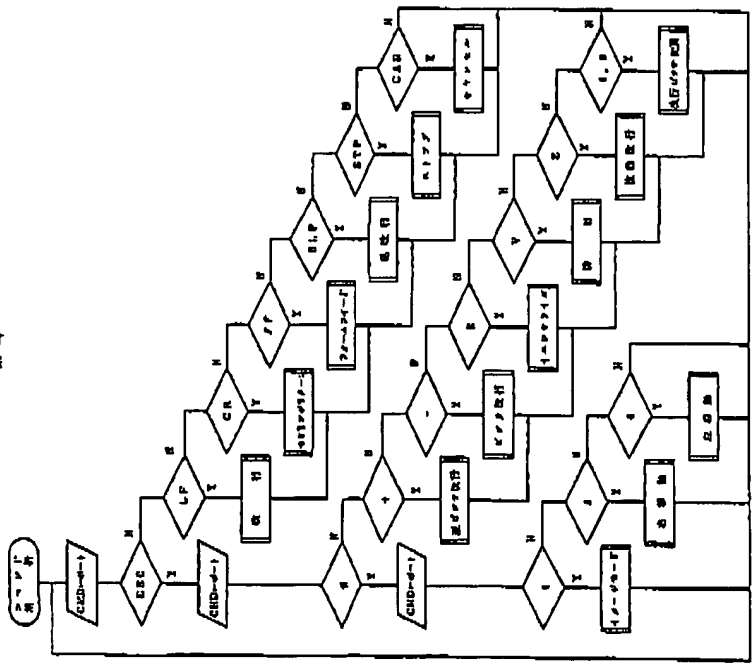
第5図



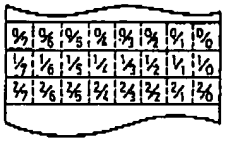
第6図



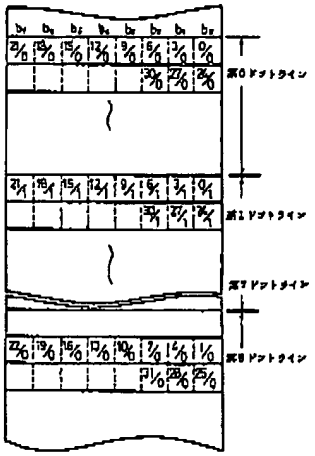
第4図



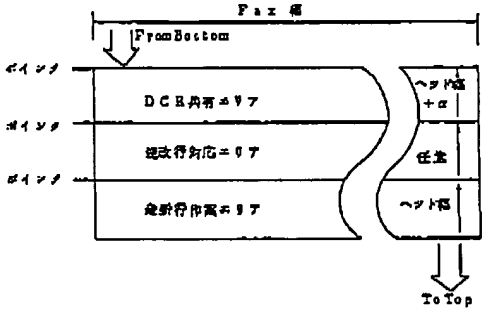
第7図



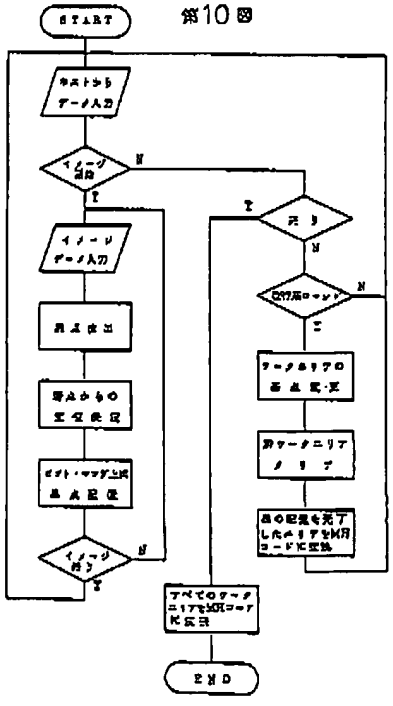
第8図



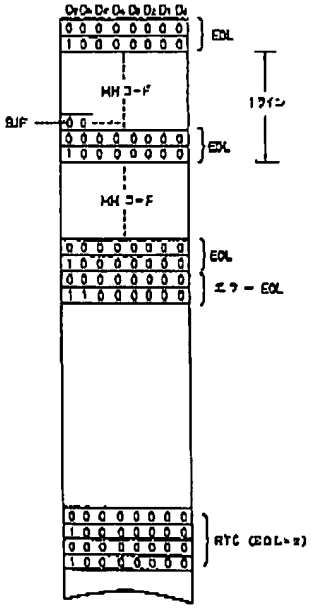
第9図



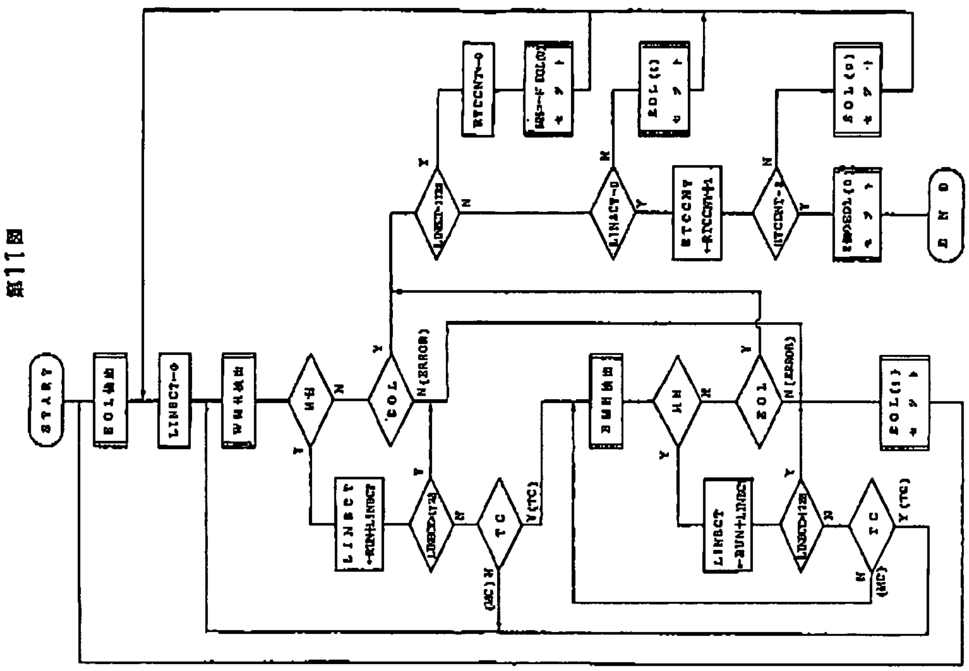
第10図



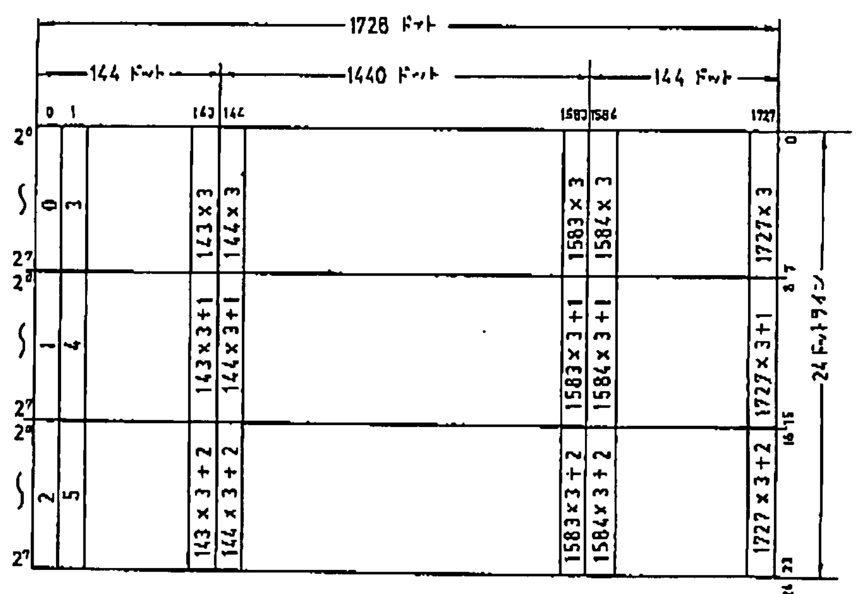
第12図



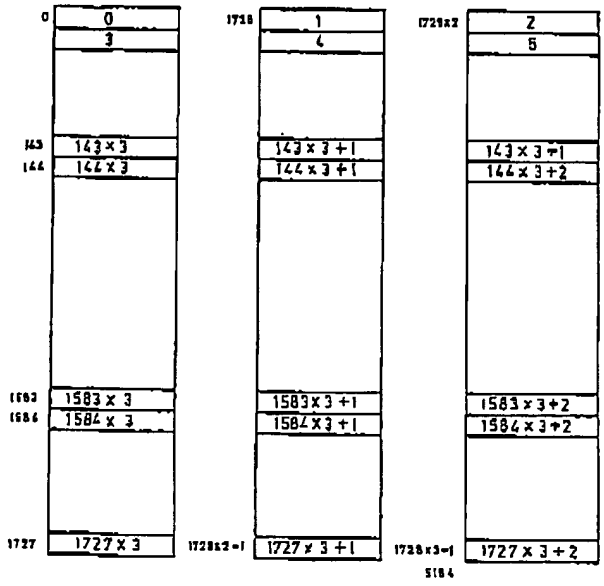
第11図


START
END
EOL
TC
Y (TC)
N (MC)
N (ERROR)

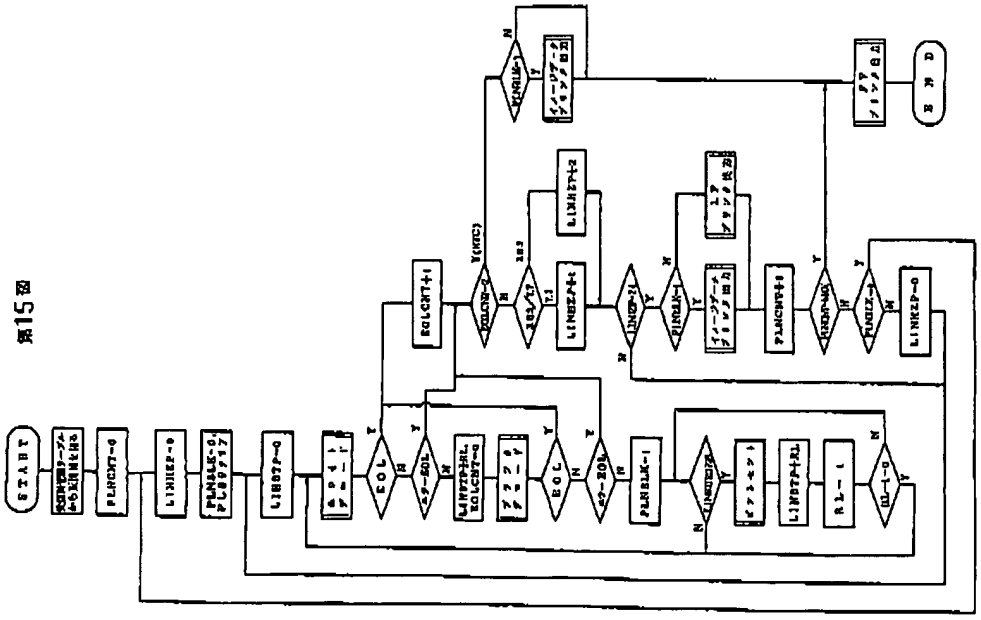
第13図


1728 ドット
144 ドット
1440 ドット
144 ドット
24ドットライン

第14図

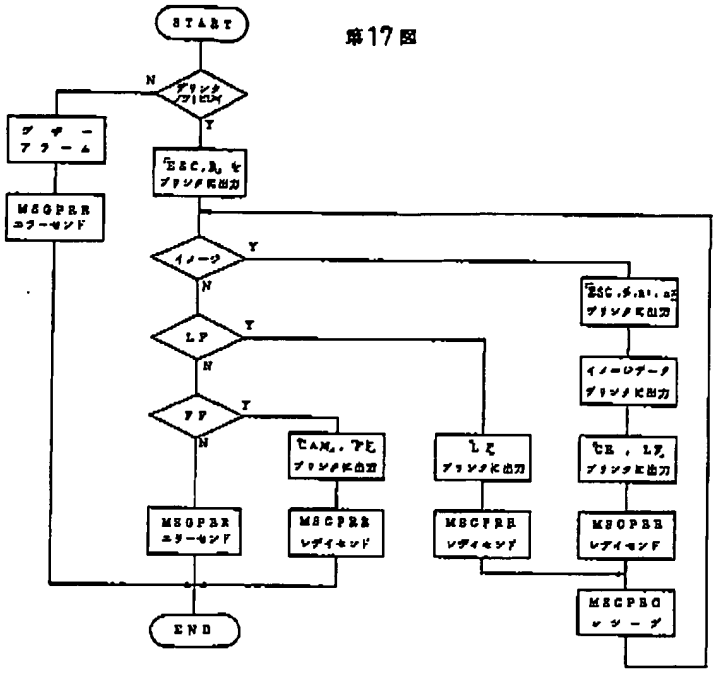

0
0
3
143
143×3
144
144×3
1583
1583×3
1584
1584×3
1727
1727×3
1728×2-1
1727×3+1
1728×3-1
5184
1727×3+2
1584×3+2
1583×3+2
144×3+2
143×3+1
1584×3+1
1583×3+1
1728×2
1728
1
4
2
5

第15図

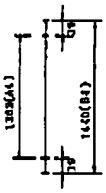


第16図

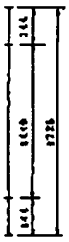
第17図



第18図



第19図



第20図